Introduction to Robotics – Exercise 3

Date of submission – Thursday 21/12/21 (23:55)

The assignment

- Use walkingOnGrid folder to Implement PRM navigation:
  1. Create a new controller (folder) named 'PRM based on 'walkingOnGrid' controller (copy the 'walkingOnGrid' controller directory and change its name to 'PRM, then – inside this directory change the name where necessary - as was shown in class).
     - implement PRM inside this controller.
  2. Create a new configuration file named 'PRM.argos' based on 'walkingOnGrid.argos' (copy the config file and change its name, then – inside change the name and path in the necessary places – as was shown in class)
     *** pay attention the call to the 'pos_and_map_loop_functions' loop function remains.
  3. Final goal: to robot should navigate successfully from starting position S = (1,1) to goal position G = (-2, -2) .
     In case there is no available path (including cases where S or G in inside or too close to obstacles) print a log to the screen stating: "No path is available" .

- You should explain your implementation preference and report its success in a PDF file
- Algorithm description - explain your implementation preferences:
  1. Which sampling did you use?
  2. How many successful samples (milestones) were sampled in total to achieve the path? (Compare at least two different quantities in your report)
  3. Did you check "same component" when adding new edges?
  4. What distance metrics was implemented? (Implement and compare at least two different distance metrics)
  5. What data structure was used for the ***k*** nearest neighbors search? (Compare at least two different ***k*** choices – e.g. 5 and 10)
  6. Explain the implantation of the local planner.
  7. What algorithm you employed for the query phase? (Dijksta, BFS, A*, etc.)

8. For all of the above components explain how it affected the road map structure and the resulted path.
9. What feedback-control you used for robot's path following?

-You can add a pseudo code if it helps you explain (but you don't have to)

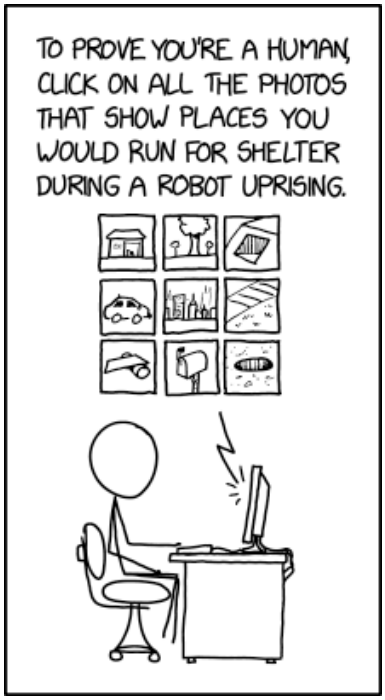
- Success report – the config file that provided to you was written such that by a simple change of the *seed* , the robot and the obstacles in the arena will appear in different places.
  Run your controller on 5 different seeds (of your choice) with successfully found paths), and for each seedreport:
  1. The seed choice (number).
  2. a screenshot of the simulation at the end of the task.
     a. The screenshot of the simulation must include the clock.
     b. At least two pairs of screenshots (four in total for each seed) for comparison between different distance metric and ***k*** configurations.
  3. A short description of the results for each seed and comparison between the screenshots.

Guidelines

- Make sure that your code is tidy and well-commented.
- Make sure that your names and IDs are listed at the beginning of every file you submit.
- Your names and IDs and any source you used should be written in a README.

What to Hand In

- You should hand-in your controller folder and a report.pdf file as described above.
- You should not hand in executable files, or any other files that can be regenerated.



GOOD LUCK! :)